IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Patent Application No. 10/526,421

Confirmation No. 3287

Applicant: Leijten, Jeroen Anton Johan

Filed: March 1, 2005

TC/AU: 2183

Examiner: Faherty, Corey S.

Docket No.: 260670 (Client Reference No. DJ/P82108US00)

Customer No.: 23460

APPELLANTS' REPLY UNDER 37 C.F.R. SECTION 41.41

Mail Stop Appeal Brief – Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

This paper is filed in response to the Examiner's Answer mailed on July 16, 2010. Appellants hereby respectfully request allowance of the pending claims for the reasons set forth in Appellants' Appeal Brief filed on April 29, 2010, and for the further reasons stated herein.

Status of Claims

Claims 1-11, 13, and 15-17 are presently pending.

Claims 1-11, 13, and 15-17 stand rejected, and these rejections are presently being appealed.

Claims 12 and 14 are canceled.

Grounds of Rejection to be reviewed on Appeal

The grounds of rejection to be reviewed on appeal are the grounds stated in the Office Action mailed on July 1, 2009. In particular, Appellant appeals:

1. The rejection of Claims 1-3, 5-7, 9, 13 and 15-17 under 35 U.S.C. §103(a) as obvious over Downing US Pat. No. 3,781,810 (Downing);

2. The rejection of Claim 4 under 35 U.S.C. §103(a) as being obvious over Downing in view of Petolino, Jr., et al. US Pat. No. 5,958,041 (Petolino);

3. The rejection of Claim 8 under 35 U.S.C. §103(a) as being obvious over Downing in view of Patterson et al., "Computer Organization & Design: The Hardware/Software Interface"(Patterson);

4. The rejection of Claim 10 under 35 U.S.C. §103(a) as being obvious over Downing in view of Forsyth US Pat. No. 5,327,566 (Forsyth); and

5. The rejection of Claim 11 under 35 U.S.C. §103(a) as being obvious over Downing in view of Lang et al., "Individual Flip-Flops with Gated Clocks for Low Power Datapaths"(Lang).

Argument in Reply to Examiner's Answer

Appellants have carefully reviewed the Answer. Section (9) of the Answer, beginning at page 3 and ending at page 10, recites the same grounds of rejection as those stated in the Final Office Action from which this Appeal is taken. Appellants have addressed the Answer's grounds for rejection in the previously filed Appellants' Appeal Brief, and thus Appellants will not repeat these arguments. For the reasons stated in Appellants' Appeal Brief and reiterated herein below in response to the Answer, Appellants seek reversal of the final rejection of each of the presently pending claims (including independent claim 1 – the only claim specifically argued on appeal).

Appellants' argument below addresses the points raised in Section 10 ("Response to Argument") of the Answer.

Appellants' arguments on appeal focus primarily upon the recited "snapshot buffer." More particularly, Appellants have argued that Downing's disclosure does not disclose or even remotely suggest the claimed snapshot buffer which "accommodates saving, ..., state information of various processor state elements, including *state information from the internal processor pipeline*." (emphasis added). Saving state information, from the internal processor pipeline, enables storing an exact state of the processor pipeline in response to an interrupt. Thus, when an interrupt is completed and the interrupted program execution sequence is restored, the program execution resumes exactly where it left off. Appellants note that their summary of (distinguishing aspects of) the invention differs from the Answers' summary that instead focuses upon: (1) the physical distinction of a snapshot buffer from a main memory and the "state information" portion of the complete term "state information from the internal processor pipeline." *See,* page 10, Section (10) Response to Argument, first paragraph, lines 1-4 and 6-8.

The Answer summarizes Downing's disclosure of a snapshot buffer that stores register data (computation results) when processing an interrupt. *See,* page 10, Section (10), Response to Argument, first paragraph, lines 4-6. Downing does not disclose saving the internal state of the processor – resulting in a need to flush any incompletely processed instruction. Downing's disclosure, when applied to a "processor pipeline" context, results in a processor design where some computation results are saved, but the contents of the internal processor pipeline are flushed and the state of execution of a command sequence "rewound" in order to ensure the integrity of the results.

The Answer asserts that "state information" is extremely generic and has been arbitrarily narrowed by Appellants. *See*, Answer, pages 10-11, Section (10), first paragraph, lines 10-22. Appellants note that the actual claim element recites "state information *from the internal processor pipeline*." (emphasis added). The actual recited element is significantly narrower than the Answer's treatment of the actual claim element and its associated context (i.e., a pipeline processor).

The Answer thereafter asserts that storing state information "for the purpose of recovery is well known in the art." *See*, Answer, page 11, lines 3-5. However, this observation regarding purported teachings in the art does not address the specifically claimed source of the saved information – i.e., "state information from the internal processor pipeline."

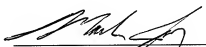
Appellants readily concede that "pipeline processor" computing architectures were well known at the time of the invention. However, Downing handles nested interrupts by saving only register data and *not* internal processor state information. *See*, Appellants' Brief, page 6, lines 4-21. Applying Downing's teachings of interrupt handling to Appellants' claimed "internal processor pipeline" architecture, results in an interrupt handling arrangement wherein contents of data registers are saved, but *state information from the internal processor pipeline* is discarded and the pipeline itself is flushed in view of the inability to re-create the internal processor pipeline's state when an interrupt occurred.

Conclusion

Downing does not disclose each element recited in Appellants' independent claim 1. The Final Office Action has not set forth a proper *prima facie* case for obviousness with regard the rejection of the presently pending claims. Appellants submit that the prior art neither discloses nor suggests modifying Downing to render the *claimed invention* wherein a snapshot buffer accommodates saving "state information from the internal processor pipeline."

For this and other reasons provided herein above, Appellants request reversal of the presently pending claim rejections.

Respectfully submitted,



Mark Joy, Reg. No. 35,562
LEYDIG, VOIT & MAYER, LTD.
Two Prudential Plaza
180 North Stetson Ave., Suite 4900
Chicago, Illinois 60601-6731
(312) 616-5600 (telephone)
(312) 616-5700 (facsimile)

Date: September 16, 2010

Claims Appendix

1. (Previously presented) A data processor comprising:

one or more functional units arranged to provide an internal processor pipeline,

one or more register files,

a data memory facility having a multibit access port facility,

a snapshot buffer, differing from the one or more register files, which during handling of an interrupt condition accommodates saving, by copying from the one or more register files to respective snapshot buffer elements, state information of various processor state elements, including state information from the internal processor pipeline, and

a controller means arranged to save, upon occurrence of a subsequent interrupt condition during handling of an actual interrupt condition, the state information of various processor state elements currently within the respective snapshot buffer elements in the data memory facility having the multibit access port facility.

2. (Previously Presented) The data processor as claimed in Claim 1, wherein said controller means are arranged to retrieve the saved contents of said snapshot buffer elements from said data memory facility through said multibit access port facility back into said snapshot buffer elements upon completing the handling of the actual interrupt condition.

3. (Previously Presented) The data processor as claimed in Claim 2, wherein said controller means are arranged to restore the retrieved saved state information of various processor state elements allowing said data processor to proceed with handling one of an earlier uncompleted interrupt or continuing a main thread of the processing.

4. (Previously Presented) The data processor as claimed in Claim 1, wherein said state information comprise latency data of current operations.

5. (Previously Presented) The data processor as claimed in Claim 1, wherein said snapshot buffer comprises output multiplexer means having said multibit access port facility for

sequentially saving selected snapshot buffer elements for transferring to said data memory facility.

6. (Previously Presented) The data processor as claimed in Claim 1, wherein said snapshot buffer comprises input multiplexer means having said multibit access port facility for sequentially selecting selected snapshot buffer elements for back-transferring from said data memory facility.

7. (Previously Presented) The data processor as claimed in Claim 1, wherein said data memory facility is operated as a stack.

8. (Previously Presented) The data processor as claimed in Claim 7, wherein said stack has a stack pointer that allows multiple stack positions per snapshot.

9. (Previously Presented) The data processor as claimed in Claim 7, wherein said snapshot buffer comprises input multiplexer means having said multibit access port facility for sequentially selecting selected snapshot buffer elements for back-transferring from said data memory facility, wherein said snapshot buffer comprises output multiplexer means having said multibit access port facility for sequentially saving selected snapshot buffer elements for transferring to said data memory facility, and wherein write and read operations in said stack are executed at mutually exclusive instants in time under control of a stack pointer.

10. (Previously Presented) The data processor as claimed in Claim 1, wherein said snapshot buffer is at least substantially constructed from shadow flipflops for storing its snapshot information.

11. (Previously presented) The data processor as claimed in Claim 1, wherein said snapshot buffer is operated at low power through one or more of clocking shadow flipflops only during actual taking of a snapshot, clocking only the shadow flipflops pointed to by a stack pointer as a top-of-stack-plus-one during a stack push operation, and clocking the stack pointer itself only during stack pointer updates that are caused by popping and pushing of a snapshot buffer stack.

12. (Canceled)

13. (Previously Presented) A data processing facility comprising the data processor as claimed in claim 1.

14. (Canceled)

15. (Previously Presented) The data processing facility as claimed in Claim 1, wherein the controller means is arranged to save the various processor state elements to the respective snapshot buffer elements in a single clock cycle.

16. (Previously Presented) The data processing facility as claimed in Claim 1, wherein the controller means is arranged to restore the various processor state elements from the respective snapshot buffer elements in a single clock cycle.

17. (Previously presented) The data processing facility of Claim 1, wherein the controller means saves, upon occurrence of the subsequent interrupt condition during the handling of an actual interrupt condition, the state information of various processor state elements currently within the respective snapshot buffer elements in the data memory facility using a stack pointer, an wherein no additional instruction bits are required for addressing the snapshot buffer elements.

Evidence Appendix
NOT APPLICABLE

Related Proceedings Appendix
NOT APPLICABLE